Today, we are going to cover loops. After we load the dataset into memory;

library(readxl)  
StudentsPerformance <- read\_excel("C:/Users/yliu3/OneDrive - Maryville University/Online DSCI502 R Programming/DataSets/StudentsPerformance.xlsx")

Your manager asks you to get the average of the math score from the scratch.

To perform this task, we may use the following for loop.

sum <- 0  
for (idx in c(1:nrow(StudentsPerformance))){  
 sum <- sum + StudentsPerformance[idx, "MathScore" ]  
}  
avg <- sum/nrow(StudentsPerformance)  
print(avg)

Note that,

* First, we initialize the sum to zero before the loop, otherwise its initial value is indeterminate
* Second, it loops over all the row numbers generated by the colon operator c(1:nrow(StudentsPerformance)
  + First iteration: idx = 1; then it takes out the math score of the 1st row and adds it to the sum in the loop body.
  + Second iteration: idx = 2; then it takes out the math score of the 2nd row and adds it to the sum in the loop body.
  + …
  + Last iteration: idx = nrow(StudentsPerformance); then it takes out the math score of last row and adds it to the sum in the loop body.
* Finally, we divide the sum by the number of rows and we get the average of the math scores **outside** of the loop.

Let’s run the codes. The average is 66.

There are two more equivalent loop structures. We may also use **while** loops to find the average of the math scores.

To perform this task, we may use the following while loop.

sum <- 0  
idx <- 1  
while (idx <= nrow(StudentsPerformance)){  
 sum <- sum + StudentsPerformance[idx, "MathScore" ]  
 idx <- idx + 1  
}  
avg <- sum/nrow(StudentsPerformance)  
print(avg)

Note here

* Before the while loop, we need to initialize the following **two** variables:
  + Sum=0, which holds the sum of the students’ math scores
  + Idx =1, which stores the current row number
* ‘while’ loop checks the condition idx <= nrow(StudentsPerformance)
  + First iteration; idx = 1 and nrow(StudentsPerformance) =1000; it becomes 1 <=1000; it is TRUE; then the while loop body is executed.
    - Add the math score of 1st row to the sum
    - Update the row number by adding 1; idx = 1+1=2
  + Second iteration; idx = 2 and nrow(StudentsPerformance =1000; it becomes 2 <=1000; it is TRUE; then the while loop body is executed.
    - Add the math score of 2nd row to the sum
    - Update the row number by adding 1; idx = 2+1=3
  + …
  + 1000th iteration; idx = 1000 and nrow(StudentsPerformance =1000; it becomes 1000 <=1000; it is TRUE; then the while loop body is executed.
    - Add the math score of 1000th row to the sum
    - Update the row number by adding 1; idx = 1000+1=1001
  + 1001th iteration; idx = 1001 and nrow(StudentsPerformance =1000; it becomes 1001 <=1000; it is FALSE; then the while loop is terminated
* Finally, we divide the sum by the number of rows and we get the average of the math scores **outside** of the loop.

Note that it is **our obligation to modify the index inside of the while loop**. If you forget to update the index, you will have an **infinite loop** since the logical condition is always TRUE. It is a typical bug for while loop.

Let’s run the codes. The average is 66.